

Tartu Ülikool
Loodus- ja täppisteaduste valdkond
Tehnoloogiainstituut

Martin Hallist

ROBOTIPÕHINE KAUGKOHALOLU KÄTE LIIGUTUSTE ÜLEKANDMISEKS

Bakalaureusetöö (12 EAP)

Arvutitehnika eriala

Juhendaja:

Robotika dotsent Karl Kruusamäe

Tartu 2019

Resümee

Robotipõhine kaugkohalolu käte liigutuste ülekandmiseks

Robootika kiire areng toob kaasa vajaduse tarkadeks robotiteks. Veel ei ole loodud lahendust, mis oleks sama kohanemisvõimeline kui inimene. Seepärast on praegune parim lahendus luua vahendid inimeste ning robotite koostööks.

Paraku ei kasutata potentsiaali piisavalt ära ning paljud robotite juhtimissüsteemid on kohmakad ning ebamugavad. See motiveerib looma süsteemi, mida oleks kerge ja mugav juhtida.

Lõputöö keskendub teiste sarnaste lahenduste ning vajalike töövahendite tutvustamisele. Peamiseks nõudeks on manipulaatori vaba liikumine. Selle tulemusena valmis kaugjuhtimise lahendus, kus robotmanipulaator liigub vastavalt käte liikumiskiirusele ning pöördenurgale.

CERCS: T120 Süsteemitehnoloogia, arvutitehnoloogia; T125 Automatiseerimine, robootika, control engineering

Märksõnad: ROS, robootika, kaugjuhtimine, Leap Motion Controller

Abstract

Teleoperation robot for arms motions

The rapid development of robotics raises the need for smart robots. There are no solutions yet that are as adaptable as human beings. Thus, the current best solution is to create solutions for collaborations of humans and robots.

Unfortunately a lot of potential is wasted, making controlling systems awkward and aggravating. It gives motivation to create a system that is easy and comfortable to maneuver.

The bachelor's thesis focuses on other similar solutions and gives an overview about used tools and modules. The main requirement is free movement of a manipulator. As the result, the teleoperation solution is made, where the robot manipulator moves according to the speed of movement of the hands and follows the angle of rotation.

CERCS: T120 Systems engineering, computer technology; T125 Automation, robotics, control engineering

Keywords: ROS, robotics, teleoperation, Leap Motion Controller

Jooniste loetelu

Joonis 1. ZEUS juhtimissüsteem [13]	11
Joonis 2. Operatsioon ZEUS kasutades [13]	11
Joonis 3. Da Vinci kirurgiarobot Xi [14]	11
Joonis 4. Da Vinci Xi juhtimissüsteem [14]	11
Joonis 5. TALON [18]	12
Joonis 6. TALON juhtimissüsteemid [19]	12
Joonis 7. Interact Centaur [26]	12
Joonis 8. Interact Centaur juhtkang [27]	12
Joonis 9. iCub juhtimine [30]	13
Joonis 10. Gazebo koordinaatteljestik	17
Joonis 11. Leap Motioni kaks infrapunakaamerat ning kolm infrapuna leedi [54]	18
Joonis 12. Leap Motion kontrolleri koordinaatsüsteem [52]	18
Joonis 13. Tööpõhimõte	20
Joonis 14. MoveIt! arhitektuur	21
Joonis 15. Gazebo maailm koos robotmanipulaatoriga UR5	22
Joonis 16. jog_arm arhitektuur	23

Lühendid, mõisted

Dublantmanipulaator (*master-slave manipulator*) – manipulaatori juhtsüsteem, mille algatusvõime on ülemaal (*master*) ning alluv (*slave*) täidab neid

Endoskoopia – õõneselundite sisevaatlus

IMS Lab (Intelligent Materials and Systems Lab) – Tartu Ülikooli tehnoloogiainstituudi arukate materjalide ja süsteemide labor

Kaugjuhtimine (*teleoperation*) – roboti juhtimise viis

Kimp (*package*) – tarkvarakogum, mis võib omakorda sisaldada teisi kimpe

Kuulutaja (*publisher*) – sõlm teemade avaldamisel

Laparoskoopiline koletsüstektoomia (*Laparoscopic Cholecystectomy*) – lihtsustatult sapipõie eemaldamine

LM-kontroller (Leap Motion Controller) – käte jälgimiseks ning tuvastamiseks kasutatav seade

Paindesensor (*flex sensor*) – sensor, millel painde suurenedes suureneb takistus

ROS (Robot Operating System) – robotikaarendusplatvorm

Sõlm (*node*) – programm, mis täidab kindlat ülesannet

Sõnum (*message*) – infokandja ROS-i sisesel suhtlusel

Telliija (*subscriber*) – sõlm teemade kuulamiseks

Sisukord

Resümee	2
Abstract	3
Jooniste loetelu	4
Lühendid, mõisted	5
1. Sissejuhatus	8
1.1 Probleemi tutvustus	8
1.2 Töö eesmärk ja ülevaade	9
1.3 Töö ülesehitus	9
2. Robotipõhise kaugkohalolu ülevaade	10
2.1 Juured	10
2.2 Tänapäev	10
2.2.1 Tervishoid	10
2.2.2 Ohtlikud keskkonnad	12
2.2.3 Meelelahutus	13
3. Töövahendite ülevaade	15
3.1 ROS	15
3.1.1 Tööpõhimõte	16
3.1.2 Ajalugu	16
3.2 Gazebo	17
3.3 Leap Motion kontrolleri	17
4. Kaugjuhitava roboti lahendus	20
4.1 Nõuded	20
4.2 Disain	20
5. Tulemused	21
5.1 MoveIt!	21
5.2 jog_arm	22
5.3 Esinenud probleemid	24
6. Tulemuste analüüs ja järeldused	26
6.1 Peamised tulemused	26
6.2 Võimalikud edasiarendused	26

Kokkuvõte	27
Viited	28
Lisad	34
Lisa 1	34
Litsents	35

1. Sissejuhatus

Sõna „robot” päritoluks peetakse tšehhikeelset sõna *robota*, mis tähendab sundtööjõudu [1,2]. Üldiselt seostatakse robotit masinaga, mis teeb etteantud monotoonseid liigutusi täpselt, kiiresti ning suure töökindlusega. Hetkel eksisteerib mitmeid kõrgtehnoloogilisi humanoidroboteid üle maailma nagu näiteks teenindusrobot UBTech Walker [3], allvee kaugjuhitav humanoidrobot OceanOne [4] või inimvõimekuse tasemel liikuv Boston Dynamicsi robot Atlas [5]. Paraku ei ole robotid veel nii võimekad kui ulmefilmides ning ei vasta kõikide inimeste ootustele. See aga ei tähenda, et see oleks saavutamatu.

1.1 Probleemi tutvustus

Roboteid kasutatakse peamiselt kindlaks otstarbeks, seetõttu on neile loodud piiratud kood. Targemate robotite loomiseks on võimalik kasutada näiteks tehisintellekti. Samas on nende õppimisprotsess äärmiselt ajamahukas, näiteks AlphaGo jaoks mängiti läbi 29 miljonit mängu [6]. Paraku ei ole võimalik kõiki olukordi läbi simuleerida, eriti situatsioonides, kus puudub defineeritud punktisüsteem või võiduvõimalus, mille järgi hinnata tehisintellekti arengut. Seepärast on mõistlik luua liitlahendus robotite ja inimeste tööks, kus varasemalt defineeritud protsessid läbib robot ning vajadusel saab inimene uutes situatsioonides kontrolli haarata.

Paraku ei pöörata kasutajaliidestele piisavalt tähelepanu ning seetõttu juhitakse roboteid kaugjuhtimispuldi või klaviatuuri abil. See muudab roboti juhtimise palju kohmakamaks ning tundub inimesele võõras. Roboti juhtimine võiks pigem toimuda erinevate sensorite abil, mis edastavad kasutaja liigutusi robotitele ning see edastab vastatest sensoritest saadud andmed kasutajale. Seetõttu on oluline roll humanoidrobotitel, mis on disainitud jäljendama inimkäitumist, näiteks käte ja pea liikumine. Palju harjumuspärasem on kuvada kasutajale sama kaamerapilt, mida näeb roboti pea ning samastada käte liikumine inimliikumisele. See tagab kiire kohanemise ning aitab vältida inimeste ja juhtsüsteemidevahelisi vigu olukordades, kus on oluline reaktsioonikiirus.

1.2 Töö eesmärk ja ülevaade

Bakalaureusetöö eesmärk on luua lahendus, mis jälgib inimese käsi ning edastab pideva liikumise robotmanipulaatorsüsteemile, mis imiteerib inimkäte liikumist loomulikul moel.

Selle tarkvaraliseks lahendamiseks pakub kõige paremat võimalust ROS (*Robot Operating System*) [7], mis laseb oma arvukaid eraldiseisvaid mooduleid kokku sobitada ühtseks tervikuks.

Terviklikku lahendust saab rakendada meelelahutussektoris, kaugjuhitavates kohalolu nõudvates situatsioonides või ohtlikes keskkondades, mis on ettearvamatud ning seetõttu nõuavad inimsekkumist.

1.3 Töö ülesehitus

Töö esimeses osas (pt 2, 3) annan ülevaate robotipõhisest kaugkohalolust ning tutvustan kasutatavaid töövahendeid. Töö kirjeldavas osas (pt 4) annan ülevaate töö suunast ning sellega seonduvatest nõuetest. Lõpuosas (pt 5, 6) analüüsin saadud tulemusi, räägin esinenud kitsaskohtadest ning arutlen võimalike edasiarenduste üle.

2. Robotipõhise kaugkohalolu ülevaade

Robotid pakuvad töökindlust ja täpsust, inimesed seevastu võimet kohaneda kiiresti uute situatsioonidega. Seetõttu on mõistlik kasutada mõlema tugevusi, et saavutada optimaalseim lahendus. Kohalolupõhise juhtimise eesmärk on täiustada inimkasutajat roboti võimekusega, samal ajal jättes mulje, et viibitakse ise situatsiooni keskmes [2].

2.1 Juured

Esimesi viiteid kaugjuhtimise alustele võib seostada Nikola Teslaga, kes juhtis 1898. aastal kaugjuhitavat paati, kasutades raadiolaineid [8,9]. 1949. aastal lõi Ray Goertzi mehaanilise dublantmanipulaatori, mida kasutati keemiliste ning radioaktiivsete ainete transpordiks ning teisaldamiseks. Hiljem, 1954. aastal, täiustati varasemalt loodud manipulaatorit, luues elektromehaanilise lahenduse. Seetõttu peetakse Goertzit tänapäevase kaugjuhtimissüsteemide rajajaks. [9]

2.2 Tänapäev

Tehnoloogia ning robotika kiire arengu tulemusena on robotid muutunud palju kättesaadavamaks. Lihtsamad komplektid on saadaval poodides, toimuvad igaaastased robotikaalased festivalid, näiteks Robotex [10] või Robolahing [11]. Lisaks üha enam uuringuid keskendub robotika kasutamisele valdkondadele, kus see polnud varasemalt kuluefektiivne või esmapilgul puudus vajadus.

2.2.1 Tervishoid

ZEUS Robotic Surgical System (edaspidi ZRSS) oli esimene kaugjuhitav robotkirurgiasüsteem [12]. Sellega tehti edukas laparoskoopiline koletsüstektoomia (lihtsustatult sapipõie eemaldamine) 68-aastasele naisele. Kirurgilise operatsiooni kaugjuhtimine toimus New Yorkist Strasbourgi, kus asus ZRSS. Operatsiooni juhtimine toimus juhtkonsooli abil (joonisel 1), mis juhtis kahte robotmanipulaatorit (joonisel 2). Lisaks edastati juhtkonsooli monitoridele endoskoopiline (õõneselundite sisevaatlus) kaamerapilt, mille kaamera liikumine toimus kaugjuhtimise kaudu. Ühenduse kiiruseks oli 10 MB/s, mille kaudu ning vastavad käsud, mediaan latentsusaeg oli 155 ms. [13]



Joonis 1. ZEUS juhtimissüsteem [13]



Joonis 2. Operatsioon ZEUS kasutades [13]

Tänapäeval üheks tuntumaks robotkirurgia vallas on Intuitive Surgical loodud da Vinci [14] robotsüsteem. 2017. aasta seisuga on üle 4400 masina müüdud üle maailma [15]. Lähimad neist asub Soomes ja Rootsis [16]. Nende mudel Xi (joonisel 3) kasutab nelja robotmanipulaatorit, mis teevad inimkäest täpsemaid liigutusi [17]. Joonisel 4 on kujutatud juhtsüsteemi, kus juhtimine toimub kahe juhtkangiga, kus 10 cm liigutusele vastab 2 cm manipulaatori liikumine. Lisaks saab juhtkangi abil muuta erinevate otsikute asendit ning pöördenurka. Manipulaatorite vahetamine, endoskoopia- ning kaamerapildi teravustamine toimub jalgpedaalide abil. Info edastamiseks kasutatakse kahte kaamerat, et luua ruumiline kaamerapilt. Lisaks on võimalus kinnitada endoskoopia- või fluoroessentskaamera manipulaatorile, mida saab vastavalt vajadusele juhtida. [14,16,17]



Joonis 3. Da Vinci kirurgiarobot Xi [14]



Joonis 4. Da Vinci Xi juhtimissüsteem [14]

2.2.2 Ohtlikud keskkonnad



Joonis 5. *TALON* [18]



Joonis 6. *TALON* juhtimissüsteemid [19]

Qinetiq robot TALON (joonisel 5) [18] on laialdaselt kasutatud ohtlikutes olukordades. Seda kasutati Fukushima tuumakatastroofi järgselt radiatsiooni mõõtmisel [20], kriisi ja sõjakolletes Afganistanis [21,22,23,24] ja Poola ning Tšehhi sõjavägi kasutab seda pommirobotina [24]. Lisaks on sellel arvukaid mooduleid, mille abil saab tuvastada ohtlikuid kemikaale, narkootikume, lõhkeaineid ning palju muud [18,23,24]. Liikumiseks on robotil roomikud, mis suudavad lähida raskeid tingimusi ning võimaldab kuni 45° treppidest üles liikuda [18,22]. Robot kasutab nelja kaamerat, kuhu saab lisada kokku 7 kaamerat [23,24]. Juhtimine toimub kuni 800 m kaugusele [18,22,23]. Juhtimissüsteemina on peamiselt kasutusel ligi 15 kg kaaluv juhtsüsteem (joonisel 6 paremal), mille juhtimine toimub kahe juhtkangi kaudu [19,25]. Lisaks on olemas juhtpuldi lahendus, mida saab kasutada nii juhtsüsteemil kui ka sülearvutitele loodud lahendusel [19].



Joonis 7. *Interact Centaur* [26]



Joonis 8. *Interact Centaur* juhtkang [27]

Interact Centaur on kulgur (kujutatud joonisel 7), millel on neli ratast ning nelikvedu, kaks jõutagastussidemega robotmanipulaatorit ning erinevad asukoha-, sügavus- ning kaugussensorid. 2015. aasta septembris toimus kulguri esmatest, kus testiti 400 km kaugusel asuvast rahvusvahelisest kosmosejaamast roboti täpsust. Eksperimendi ülesandeks oli simuleerida olukorda, kus tuleb teisel planeedil ühendada juhtmeid. Selleks tuli asetada metallvarras 4 cm sügavusele avasse, kus liikumisruumi oli kõigest 150 μm . Täpsuse tagamiseks kasutati jõutagastussidemega juhtkangi (kujutatud joonisel 8), et kasutaja tunneks takistust liikumisel. Esimene katse õnnestus 45 minutiga, järgmine juba 10 minutiga, mis näitab kiiret õppimis- ning kohanemisvõimet kontrollseadmega. Roboti võimalikud kasutusala ei piirdu ainult kosmosega, seda on võimalik rakendada ka oludes, mis on ebasobivad inimeste jaoks. [28,29]

2.2.3 Meelelahutus



Joonis 9. *iCub juhtimine* [30]

2015. aastal 15. IEEE-RAS humanoidrobotite konverentsil tutvustatud teadustöös [30] keskenduti iCub [31] roboti juhtimisele. Selleks kasutati paindesensoritega kindaid SensorClove [32], Oculus Rifti [33] ning Microsoft Kinecti [34], mille abil juhiti nii roboti

pead kui ka käsi (kujutatud joonisel 9). Lõpptulemusena oli võimalik liigutada iCubi pead ning kaela ning tema käte abil võtta üles esemeid ning paigutada kasutaja juhitud kohta.

Teine sarnane näide on humanoidrobot Nao [35] juhtimisest. 2012. aastal loodud teadustöös [36] keskenduti samuti robotite liigutustele, aga võrreldes eelmise robotiga, oli lisatud ka jalgade juhtimine. Roboti juhtimine toimus Microsoft Kinecti [34] abil, millega määrati pea, käte ning jalgade liikumine. Tulemusteks olid lihtsamad manipulatsioonid pea, käte ning jalgade juhtimiseks.

Saadud tulemusi on võimalik rakendada meelelahutussektoris, kus üheks kasutuselaks oleks vahemaa tagant suhtlus ning kontakti hoidmine. Näiteks on võimalik vanaemal mängida lapselapsega, kellel muidu ei ole võimalik distantse tõttu lapselapsega kokku puutuda. Samuti võimaldaks robotit Nao kasutades läbi viia lihtsamaid treeninguid ilma kohal viibimata.

3. Töövahendite ülevaade

ROS laseb keskenduda kaugjuhtimissüsteemi loomisele võimalikult efektiivselt, sest see laseb kasutada juba varasemalt eksisteerivatele kimpe. Peamiseks ülesandeks jääb nende kokkusidumine ühtseks tervikuks ning juhtimisloogika väljatöötamine. Lisaks on oluline ROS-iga kaasas olev tööriist Gazebo, mis võimaldab luua simulatsiooni kogu lahendusest. Läbi LM kontrolleri toimub käte tuvastamine, seetõttu on oluline tutvustada ka selle toimimise aluseid.

3.1 ROS

ROS on tarkvarakogum erinevatest tööriistadest ja tekidest, mis aitavad lihtsustada robotite kasutamist. See on vabavara, kus panustatakse ühiselt erinevate väiksemate tarkvaramoodulite arendamisele. Nii saab juba varem loodud koodi taaskasutada, kohendada enda tingimustele vastavaks ning hiljem jagada ka teistele. [37,38]

ROS vajab toimimiseks Linux-tüüpi operatsioonisüsteemi, nt Ubuntu. ROS-i puhul on tegemist pigem tarkvaraga, mis käitub justkui operatsioonisüsteem. Selle peamiste tööülesannete hulka kuulub madalatasemeliste seadmete kontrollimine ja juhtimine, protsesside vaheline suhtlus, paketiholdus, riistvara abstraktsioonikihtide tagamine ning erinevate funktsioonide rakendamine. [38]

ROS laseb luua tarkvaralahendusi erinevates programmeerimiskeeltes – praegu toetab Pythonit, C++ ja Lispi, samuti on eksperimentaalsed teegid Java ja Lua toetamiseks. Samuti on koodi lihtne siduda teiste robotijuhtimist võimaldavate tarkvaradega. Praegu on ROS-i kasutades võimalik siduda rakendusi näiteks OpenRAVE, Orocos ja Player. [38]

ROS töötati välja ideega, et teha asjad võimalikult lihtsasti kasutatavaks ning seetõttu on nähtud palju vaeva vastava infolehega nende wikis [39]. Sealt võib leida seadmete kirjelduse, seadme teekide haldavate isikute või üksuste kontaktandmed, parameetrite selgitused ning palju muid suunavaid linke või allikaid seadmete ja tarkvaraga tegelemiseks.

3.1.1 Tööpõhimõte

Peamine ROS-i tööpõhimõte seisneb sõlmede (*node*) [40] koostööl, mis lahendavad erinevaid ülesandeid. Sõlmed vahetavad omavahel andmeid, mida nimetatakse sõnumiteks (*messages*) [41]. Sõnumitel on lihtne andmestruktuur, kus on msg-tekstifailides defineeritud sõnumi tüüp, näiteks täisarv, ujukomaarv või tõeväärtus. [42]

Sõnumeid saadetakse välja kuulutaja (*publisher*) ja tellija (*subscriber*) põhimõttel. See tähendab, et sõnumid jõuavad määratud rubriikidesse (*topic*) [43]. Avaldajad ja jälgijad teavad ainult rubriike, mitte üksteise olemasolu. Üks sõlm võib mitut erinevat sõnumit saata ning samuti tellida informatsiooni mitmelt sõlmelt korraga. Üldine tööpõhimõte on analoogne DNS-serveriga. [42]

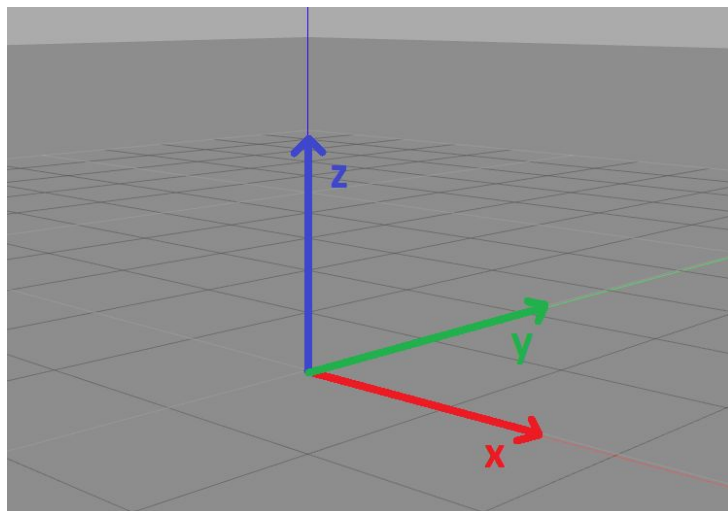
Teenused (*service*) [44] koosnevad kahest eri tüüpi sõnumist ehk päringust ja vastusest, mis töötavad paarina. Erinevus sõnumite ja teenuste vahel on see, et sõnumit saab saata ühele kindlale sõlmele, mitte ei edastata seda kõigile korraga. Informatsiooni edastamiseks on vaja teada edastatava sõlme nime ning pärast saatmist jäädakse ootama vastust. [42]

3.1.2 Ajalugu

ROS sai alguse 2007. aastal, kui Stanfordini ülikoolis töötati välja robot nimega STAIR (Stanford AI Robot) [45]. Selle roboti loomisel liideti kokku sellel ajal olnud uudsed tehnoloogilised lahendused. Nendeks olid masinõpe, kaamerapildi analüüs, navigatsioon, asjade ümberpaigutamine robotkäe abil ning otsuste tegemine kui ka kõnetuvastus [45,46]. Idee oli luua midagi uudset, mida saaks kasutada väga mitmekülgselt nii kodus kui ka kontoris [45]. Willow Garage otsustas ideega edasi minna ning kulutas palju aega, et testida läbi erinevaid rakendusi. Sarnaseid projekte loodi mujalgi ning järk-järgult hakati avaldama valmis robotite koodi, mis lisati ROS-i. Willow Garage juhtis ROS-i kuni 2013. aastani, kuni Open Source Robotics Foundation jätkas ROS-i haldamise ning edasiarendamisega. [47]

Praeguseks hetkeks on olemas 12 ROS-i distributsiooni, mis on seotud loomise hetkel peamises kasutuses oleva Linux-operatsioonisüsteemidega. Hetkel on veel tugi olemas neljale ROS-i distributsioonile, mida järjest uute Linux-süsteemide tulekuga välja vahetatakse. [48]

3.2 Gazebo

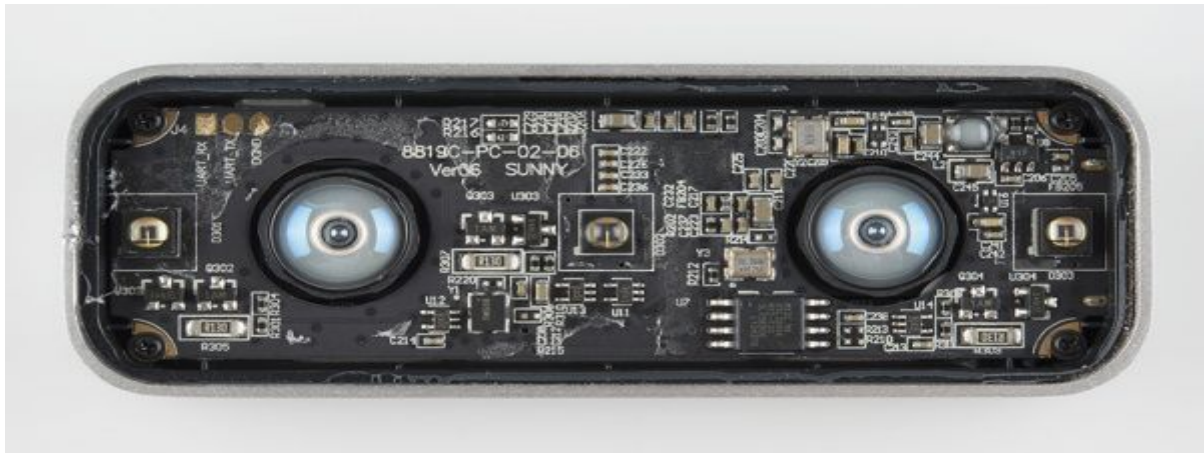


Joonis 10. *Gazebo koordinaatteljestik*

Gazebot võiks pidada üheks olulisemaks ROS-i lisaks juba seetõttu, et igal uuel ROS-i väljalaskel valitakse ka Gazebo versioon. Indigol on see 2. versioon, Kineticul ja Lunaril 7. versioon ning Melodicul 9. versioon [49]. See on tööriist, mis laseb luua simulatsioone ning testida roboteid. Sellega on võimalik testida erinevaid sensoreid, robotite liikumist või programmi töötavust enne reaalse robotil kasutamist. Eriti mugavaks teeb asjaolu, et sellel on olemas mitmeid füüsikalisi parameetreid, mis on üsna võrreldavad päriseluga, näiteks kokkupõrgetel põrkub robot impulsi tõttu tagasi [50]. Objektide maailma paigutamine toimub joonisel 10 kujutatava koordinaatteljestiku põhjal.

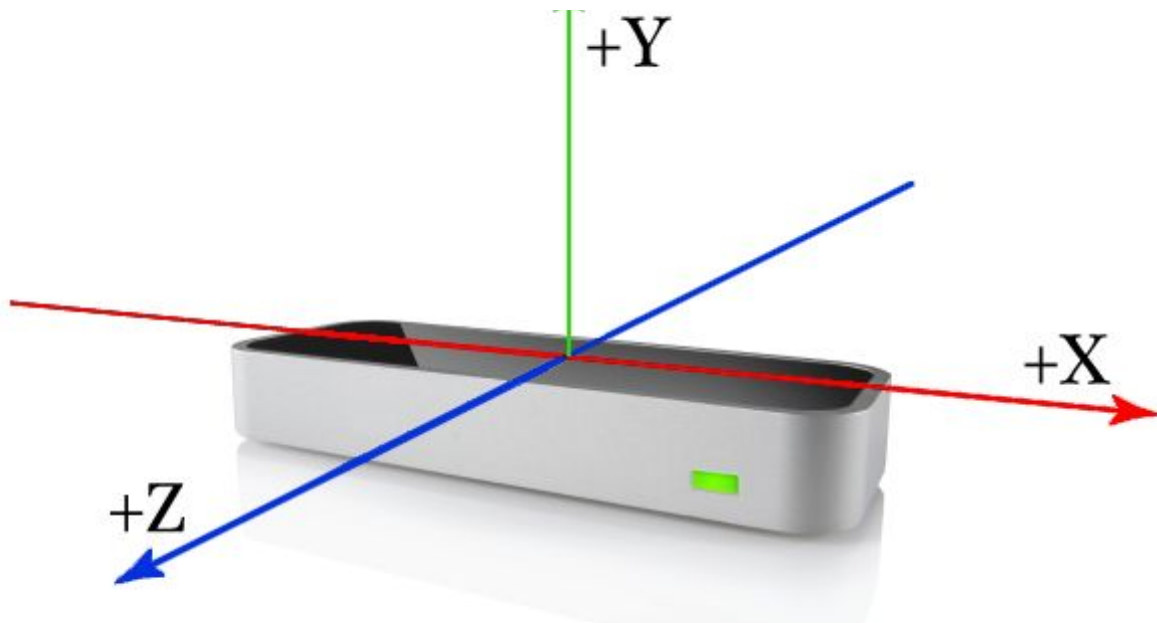
3.3 Leap Motion kontrolleri

Leap Motion kontrolleri [51] kasutab käeliigutuste tuvastamiseks kaht infrapunakaamerat ning kolme infrapunaandurit (kujutatud joonisel 11 ruudukujulistes kastides kaamerate kõrval). Kaamerate nägemisulatus on ligikaudu 150 kraadi ning efektiivne lugemiskõrgus seadme kohalt 25–600 millimeetrit. Peamine kauguse piiraja on vool, mille LM-kontroller saab USB-ühenduse kaudu. Infrapunaemitterites kasutatakse 850 nm lainepikkust infravalguse kuvamiseks ning kaamerad tuvastavad kätelt tagasipeegeldunud valguse ning sellest moodustab omakorda 3D-mudeli. [52,53]



Joonis 11. Leap Motioni kaks infrapunakaamerat ning kolm infrapuna leedi [54]

Saadud 3D-mudelist leitakse kõige tõenäolisem positsioon käte ja sõrmede asukohast ruumis, samuti järeldatakse võimalikud peidetud käeosade asukohad. LM-kontroller suudab tuvastada näiteks kasutaja käte ja sõrmede asukoha, selle asendi, erinevad luulülide positsiooni ning lihtsamaid liigutusi. [52,53]



Joonis 12. Leap Motion kontrolleri koordinaatsüsteem [52]

LM-kontroller annab käte asukoha allpool joonisel 12 kujutatud koordinaatteljestikus. Nullpunkt on defineeritud sensori keskkoha. Andmete edastamiseks kasutab LM-kontroller kauguse jaoks millimeetrid, aja jaoks millisekundeid, kiiruse jaoks millimeetreid sekundis ning nurkade jaoks radiaane. [52]

Selle võimalikuks alternatiiviks on näiteks Microsoft Kinect [34], mis suudab tuvastada peamisi kehas olevaid lülisid. Sensoritena kasutakse värvikaamerat ning sügavussensorit. Püsti seisval inimesel suudab see määrata kokku 25 erinevat osa, mis muudetakse mudeliks. Kinect sobib peamiselt kasutamiseks mõjuri (*end-effector*) juhtimiseks, kus ei ole oluline täpsus ning robotmanipulaatori lülide eraldiseisev juhtimine. Nimelt on nihe reaalse asukohaga 50–100 mm ning tuvastatakse randme asukoht ning sõrmede ots [55]. LM-kontrolleril seevastu ~0,5 mm nihkega ning suudab lisaks tuvastada peamised luud käelabas [56].

4. Kaugjuhitava roboti lahendus

Töö eesmärgiks on luua vahelüli andmevahetusele. See tähendab, et loodav lahendus töötleb ümber saadud andmed ning luuakse juhtloogika ROS-i kimpude vahel, mis vahendab andmeid sobival kujul Leap Motion kontrolleri ning robotmanipulaatori vahel.

4.1 Nõuded

Loodav süsteem peab olema võimalikult sarnane inimliikumisele. See tähendab, et liigutuste imiteerimine peab olema pidev ning märgatava viivitusega. Lisaks sellele peaks seda olema mugav kasutada, ilma piiravate faktoriteta, näiteks käe liikumise tuvastamiseks kasutatava seadme juhtmed. Samuti sisaldab see piirangud käte võimalike liikumiste osas – käte liikumine toimub ainult inimese ees.

4.2 Disain



Joonis 13. Tööpõhimõte

Disaini poolest on loodud lahendus kõigest vahelüli, mille tööpõhimõtet on kujutatud joonisel 13. Leap Motion kontrolleri ülesandeks on edastada info käte olemasolust. Saades vaste sobivast käest küsitakse olemasolevate käe asukohta koordinaatteljestikus ning kaldenurka. Saadud andmed algselt kasutavad joonisel 12 kujutatud koordinaatteljestikku, mis muudetakse Gazebos kasutusel olevasse süsteemi, mis on joonisel 10.

Saadud andmed läbivad juhtloogika, kus saadud informatsiooni põhjal pannakse paika liikumissuund xyz-teljeliste kiirusvektorite kaudu. Samuti määratakse ära telgede kaldenurkade pöördevektorid. See omakorda saadetakse robotmanipulaatorile, mis liigub saabunud andmete põhjal.

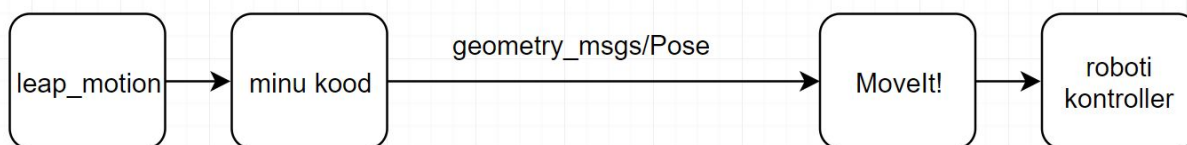
5. Tulemused

Testimiseks vajalikud baaskomponendid:

- Linux 16.04 [57]
- ROS Kinetic Kame [58]
- leap_motion ROS kimp [59]
- Leap Motion SDK v2.3.1[60]
- leap_motion kasutamiseks vajalik leaped-teenus [61]
- universal_robot [62]
- MoveIt! ROS kimp [63]

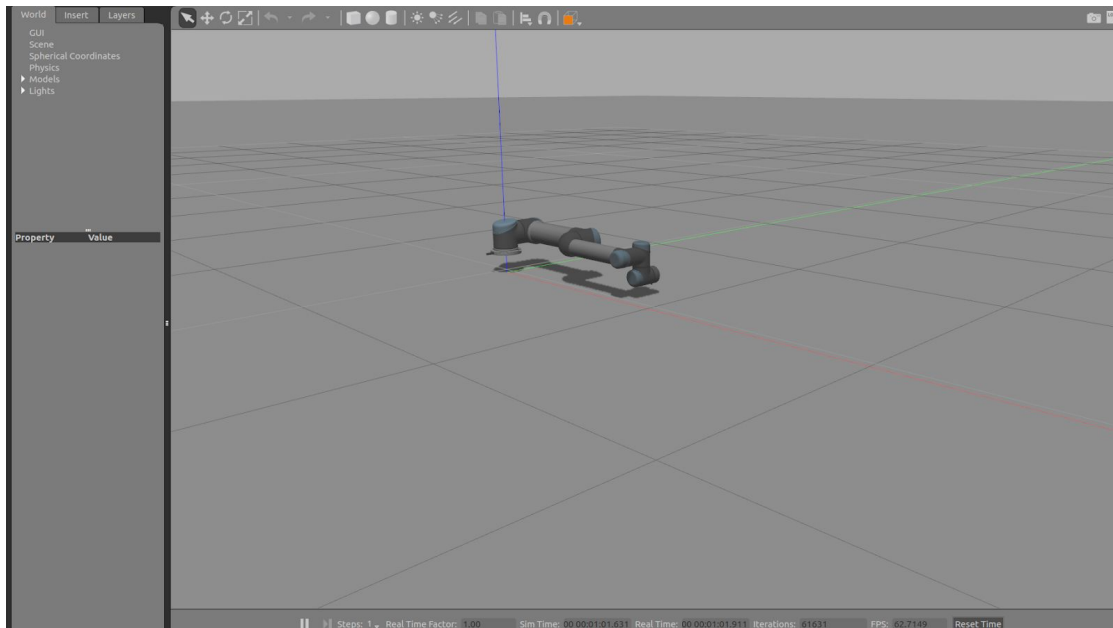
5.1 MoveIt!

Testimine algas eraldiseisvate vajalike baaskomponentide testimisega, et selgitada välja nende piirangud ning sobivus antud ülesande jaoks. Esmane testimine algas universal_roboti ja MoveIt! kimpu kasutamisega. Pärast tööpõhimõtete endale selgeks tegemist testisin eraldiseisvalt leap_motion kimpu, kuhu oli vastavalt juhendile vaja ka Leap Motion SDK arendustarkvara. Paraku ei õnnestunud saada ühendust LM-kontrolleriga ning leidsin, et puudub vajalik leaped-teenus, mis vahendab andmeid. See tuli ise luua juhendi põhjal.



Joonis 14. *MoveIt! arhitektuur*

Viimaks lisasin baaskomponendid kokku, mille tööpõhimõte on kujutatud joonisel 14. See tähendab, et minu kood tellib leap_motionilt andmeid ning muudab need ümber geometry_msgs/Pose-tüüpi andmeteks. Need omakorda edastatakse MoveIt!-ile, mille kaudu toimub roboti simuleerimine Gazebo keskkonnas (kujutatud joonisel 15).



Joonis 15. *Gazebo maailm koos robotmanipulaatoriga UR5*

Üsna kiiresti testimise käigus selgus, et MoveIt! ei ole sobiv vahend sujuva liikumise saavutamiseks, isegi kui muuta MoveGroupi parameeter „setPlanningTime” võimalikult väikeseks. Probleemideks oli katkendlik liikumine, muutes liikumise liiga ebaühtlaseks ning aeglaseks.

5.2 jog_arm

Testimiseks lisandunud komponendid:

- jog_arm [64]
- keyboard_publisher [65]
- ur5_jog_arm [66]
- IMS Lab jog_arm [67]
- ur_modern_driver [68]
- Universal Robots robotmanipulaator UR5 [69]

Järgnenud testimisel otsustasin kasutada MoveIt!-i asemel jog_arm kimpu, mis leitud allikate ning videote põhjal peaks sobima palju paremini pidevaks jälgendavaks liikumiseks. Paraku oli teemakohase info kättesaadavus piiratud.



Joonis 16. *jog_arm* arhitektuur

Peamine erinevus `MoveIt!`-i ja `jog_arm` arhitektuuris (joonised 14 ja 16) on saadetavas andmetüübis. `MoveIt!`-i eelistatud andmetüübiks on „`geometry_msgs/Pose`”, kus liikumine määratakse koordinaatide ning kaldenurkadega, `jog_arm` puhul seevastu kasutab „`geometry_msgs/TwistStamped`” andmetüüpi ehk edastatakse teljeline vektori kiirus ühikuga m/s ning radiaanide pöördenurk rad/s.

`jog_arm` kimbu testimiseks rakendati `keyboard_publisher` ja `ur5_jog_arm` kimpu. Esmane testimine toimus klaviatuuri vahendusel Gazebo keskkonnas. Kasutamisel oli probleemiks singulaarsusele lähenemisel toimuv robotkäe täielik aeglustumine kuni seiskumine mõne sekundi jooksul. Abi polnud ka eelnevast liikumissuunast eemaldumise katsetest. Singulaarsus tähendab seda, et tekib olukord, kus robot ei suuda hoida defineeritud trajektoori või liikumiskiirust igas punktis [61]. Samuti ei aidanud `jog_arm` tutvustaval lehel parameetreid „`lower_singularity_threshold`” ning „`hard_stop_singularity_threshold`” vahe võimalikuks suureks muutmise.

Juhendaja soovitusel sai asendatud `jog_arm` kimp IMS Labi `jog_arm` kimbuga, mille varasemad tulemused olid teada. Esmane testimine toimus koos `keyboard_publisher` ja `ur5_jog_arm` kimbuga Gazebo keskkonnas, mis võimaldas ühe telje kaupa juhtimist. Kimbu `ur_modern_driver` ning Universal Robots robotmanipulaatori UR5 abil sai kindlaks tehtud, et `jog_arm`-i on võimalik rakendada ka reaalsel robotmanipulaatoril.

Järgmiseks sätestati `jog_arm` seadistused, et milles määrati ära juhitava lüli, maksimaalsed kiirused, avaldamisperioodi, avaldamisperioodi, singulaarsusega seotud sätted ning juhitev seadme. Edasine testimine toimus simulatsioonikeskkonnas Gazebo, mille abil toimus juhtimisloogika loomine. Edasi loodi uue seadistuste fail reaalse robotmanipulaatori

kasutamiseks jaoks ning seejärel läbi testimise sai kindlaks määratud optimaalne kiirus jog_armi sätete jaoks. Samuti prooviti kahe käega juhtida erinevaid robotmanipulaatori lülisid, mis töötas probleemivabalt.

5.3 Esinenud probleemid

Kuna jog_arm kasutab osaliselt MoveIt! funktsionaalsusi, siis on problemaatiline lülidel kogu liikumisulatuses $[-2\pi, 2\pi]$ liikumine ning see on piiratud $[-\pi, \pi]$. See tähendab, et $-\pi$ või π piiri ületades edasiliikumine ei toimu otse vaid liikumine toimub tagasi. Näiteks π -st positiivne liikumine x radiaani võrra tähendab, et lüli liigub positsiooni $-\pi+x$.

LM-kontrolleri töö vahepeal seiskub ning seepärast tuleb aeg-ajalt leapd-teenus taaskäivitada. Esineb ka teenuse seiskumist. Eelnevaid vigu esineb mõnetunnise kasutamise jooksul korra või paar. Tüüpilisem on andmeedastuse ajutine seiskumine sekundiks või paariks. Põhjuseks on tõenäoliselt LM-kontrolleri ülekuumenemine, sest see läheb töötamise käigus üsna kuumaks.

Robotmanipulaatori Gazebo launch-faili lisamine üldisesse launch-faili tekitab olukorra, kus robotmanipulaator kukub aeglaselt maha. Vea esimene pool on ROS Kineticul kasutusel oleva Gazebo 7 gravitatsiooniviga [70], mille parandus on loodud Gazebo 9-le, mida on kasutab ROS Melodic. See tähendab, et gravitatsiooni mõjust tulenev kukkumine toimub väga aeglaselt, tüüpiliselt mitme minuti jooksul. Vea teine pool on seotud sellega, et Gazebo ei jõua faile piisavalt kiiresti ära laadida ning seetõttu robot hakkab kukkuma enne, kui laetakse ära Gazebo maailma ja robotmanipulaatori baasi vaheline kinnituslüli. Seda viga ei esine, kui esmalt eraldiseisvalt käivitada robotmanipulaatori Gazebo launch-fail ning seejärel kõik muu.

Päris robotit kasutades ilmnes, et UR5 robotmanipulaator on mõeldud peamiselt ühe telje kaupa liikumiseks ning kindlast kiirusest alates muutub liikumine hüplikuks ning võib kahjustada manipulaatorit. Madalamate kiiruste juures oli liikumine sujuv ning probleeme ei esinenud.

Arendamistegevuse käigus jäi poolikuks kaht kätt kasutava süsteemi arendus, mida testiti Gazebo keskkonnas robotiga Pepper [71,72]. Peamiseks kommistuskivi oli mitmemanipulaatorilise süsteemi kasutamisel andmete edastamises. Sõlmede ning rubriikide vahelised ühendused olid loodud lahenduse, jog_armi ja vahendava lüli move_groupi vahel

loodud ning sarnanesid eelnevalt loodud üht manipulaatorit kasutavale süsteemile. Tõenäoliselt oli viga tingitud sellest, et move_group andmeid üritati saata ning vastu võtta mitteeksisteerivast rubriigist.

6. Tulemuste analüüs ja järeldused

6.1 Peamised tulemused

Lõpptulemusena tulemusena valmis lahendus Leap Motion kontrollert kasutades juhtida robotmanipulaatorit UR5. Liikumiste matkimine toimib nii Gazebo simulatsioonikeskkonnas kui ka reaalses, st et jälgitakse käte asendit ruumis. Samuti jälgendatakse asendist tulenevat kaldenurka. Kogu töö tulemus on olemas GitHub repositoorimis, mis on lisa 1 asuval lingil.

6.2 Võimalikud edasiarendused

Lisa sensori kasutuselevõtt muudaks liigutused palju täpsemaks. See aitab vältida olukordi, kus Leap Motion kontroller ajutiselt töö lõpetab ning seeläbi ka robotmanipulaator seiskub. Lisaks saab andmete keskmistamise teel leida täpsem käe positsioon. Vastavalt positsioonile saaks selle abil määrata erinevaid käe osasid, mis jäävad varju näiteks mõne kehaosa või muu objekti taha.

Töös ei realiseeritud kaht manipulaatorit kasutavat süsteemi, mida oleks väga vaja liigutuste ning mitut kätt nõudvate tegevuste jaoks. Selle loomise jaoks on tarvis läbi käia kimpudes loodud lahendused. Alternatiivseks võimaluseks on kasutada näiteks kahte eraldiseisvat robotmanipulaatorit, mis peaks töötama sarnaselt ühe manipulaatori mitme lüli juhtimisele. Seda paraku ei olnud võimalik testida, sest puudus ligipääs teisele robotmanipulaatorile.

Kokkuvõte

Robotika kiire areng loob vajaduse tarkadeks robotlahendusteks. Paraku ei ole olemas veel sobivat tehnoloogilist lahendust, mis oleks täielikult iseseisev. Seepärast on mõistlik täiustada inimest robotite võimekusega.

Töö eesmärgiks oli luua robotkäte juhtimissüsteem, mis suudab imiteerida inimliikumist. Eesmärgi saavutamiseks kasutati ROS keskkonda, Leap Motion Controllerit, jog_arm ning MoveIt! kimpu. Testimine toimus Gazebo simulatsioonis kui ka Universal Robots robotmanipulaatoriga UR5.

Mõlemal juhul saavutati ootuspärane tulemus. Loodud lahendus tuvastab käte liikumise kiirust, asukohta ning pöördenurga, mille robotmanipulaator paralleelselt järgi sooritab. Seda saab teha nii Gazebo simulatsioonikeskkonnas kui ka kasutades reaalist robotmanipulaatorit UR5.

Viited

- [1] „Science Diction: The Origin Of The Word ‘Robot’“. NPR.Org. Võrgumaterjal. Vaadatud 13. mai 2019.
<https://www.npr.org/2011/04/22/135634400/science-diction-the-origin-of-the-word-robot>
- [2] Rossi, C. ja Ceccarelli, M. 2012. RAAD 2012: 21th International Workshop on Robotics in Alpe-Adria-Danube Region, 10-13 September 2012, Naples, Italy : Proceedings. lk 18.
- [3] „Walker“. UBTECH Robotics. Võrgumaterjal. Vaadatud 29. aprill 2019.
<https://ubtrobot.com/pages/walker>
- [4] „Stanford Robotics Lab“. Võrgumaterjal. Vaadatud 29. aprill 2019.
<https://cs.stanford.edu/group/manips/ocean-one.html>
- [5] „Atlas | Boston Dynamics“. Võrgumaterjal. Vaadatud 29. aprill 2019.
<https://www.bostondynamics.com/atlas>
- [6] Silver, David, Schrittwieser, Julian, Simonyan, Karen, Antonoglou, Ioannis, Huang, Aja, Guez, Arthur, Hubert, Thomas jt „Mastering the Game of Go without Human Knowledge“. Nature 550, nr 7676 (oktoober 2017), lk 354-359.
<https://doi.org/10.1038/nature24270>
- [7] „ROS.Org | Powering the World’s Robots“. Võrgumaterjal. Vaadatud 25. aprill 2019.
<http://www.ros.org/>
- [8] Nikola, Tesla. 1898. Method of and apparatus for controlling mechanism of moving vessels or vehicles. United States US613809A.
<https://patents.google.com/patent/US613809A/en>
- [9] Li, Zhijun, Xia, Yuanqing ja Su, Chun-Yi. 2015. Intelligent Networked Teleoperation Control. Springer, lk 1-2.
- [10] „Robotex International • Biggest Robotics Festival on the Planet“. Võrgumaterjal. Vaadatud 18. mai 2019. <https://robotex.international/>
- [11] „Robolahing | Tartu | Robolahing“. Võrgumaterjal. Vaadatud 18. mai 2019.
<https://www.robolahing.ee/>

- [12] Choi, Paul J, Oskouian, Rod J, ja Tubbs, R. Shane. Võrgumaterjal. „Telesurgery: Past, Present, and Future“. Cureus 10 (5). Vaadatud 17. mai 2019.
<https://doi.org/10.7759/cureus.2716>
- [13] Marescaux, Jacques, Leroy, Joel, Rubino, Francesco, Smith, Michelle, Vix, Michel, Simone, Michele ja Mutter, Didier. 2002. „Transcontinental Robot-Assisted Remote Telesurgery: Feasibility and Potential Applications“. Annals of Surgery 235 (4). lk 487–492.
- [14] „Intuitive | Robotic Assisted Systems | da Vinci Robot“. Intuitive Surgical. Võrgumaterjal. Vaadatud 18. mai 2019.
<https://www.intuitive.com/en-us/products-and-services/da-vinci/systems>
- [15] „Company“. Intuitive Surgical. Võrgumaterjal. Vaadatud 18. mai 2019.
<https://www.intuitive.com/en-us/about-us/company>
- [16] Vendittelli, Marilena. „Da Vinci® Surgical System“. Medical Robotics. Võrgumaterjal. Vaadatud 18. mai 2019.
- [17] „Fact Sheet for da Vinci® Surgery“. Võrgumaterjal. Vaadatud 18. mai 2019.
[https://www.steliz.org/www-seb/media/SEB-PDF-Documents/da_Vinci_Surgery_Facts_Sheet_191548\(1\).pdf](https://www.steliz.org/www-seb/media/SEB-PDF-Documents/da_Vinci_Surgery_Facts_Sheet_191548(1).pdf)
- [18] „Talon, QinetiQ Robots | QinetiQ NA“. Võrgumaterjal. Vaadatud 19. mai 2019.
<https://qinetiq-na.com/products/unmanned-systems/talon/>
- [19] TALON® IV Engineer. Wired. Võrgumaterjal. Vaadatud 19. mai 2019.
https://www.wired.com/images_blogs/dangerroom/files/DS08-126-TalonIVEngineer.pdf
- [20] Ohno, K., Kawatsuma, S., Okada, T., Takeuchi, E., Higashi, K. ja Tadokoro, S. 2011. „Robotic control vehicle for measuring radiation in Fukushima Daiichi Nuclear Power Plant“. 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics, 38–43. Võrgumaterjal. Vaadatud 19. mai 2019.
<https://doi.org/10.1109/SSRR.2011.6106792>
- [21] QinetiQ’s TALON Now in Use in Afghanistan. Military Technology. 2010. 34 (9): 15-16.
- [22] TALON Datasheet. QinetiQ NA. Võrgumaterjal. Vaadatud 19. mai 2019.
https://qinetiq-na.com/wp-content/uploads/TALON5_Datasheet_LR.pdf

- [23] TALON HAZMAT Datasheet. QinetiQ NA. Võrgumaterjal. Vaadatud 19. mai 2019. https://qinetiq-na.com/wp-content/uploads/TALON_HAZMAT_LR.pdf
- [24] „TALON Tracked Military Robot“. Army Technology. Võrgumaterjal. Vaadatud 19. mai 2019.
<https://www.army-technology.com/projects/talon-tracked-military-robot/>
- [25] „TALON Small Mobile Robot“. Võrgumaterjal. Vaadatud 19. mai 2019.
<https://www.globalsecurity.org/military/systems/ground/talon.htm>
- [26] „The Interact Centaur Rover“. European Space Agency. Võrgumaterjal. Vaadatud 18. mai 2019.
https://www.esa.int/spaceinimages/Images/2015/09/The_Interact_Centaur_Rover5
- [27] „Image: Controlling Robots at the Human Robot Interaction Laboratory“. Võrgumaterjal. Vaadatud 19. mai 2019.
<https://phys.org/news/2017-05-image-robots-human-robot-interaction.html>
- [28] „Slam Dunk for Andreas in Space Controlling Rover on Ground“. European Space Agency. Võrgumaterjal. Vaadatud 18. mai 2019.
https://www.esa.int/Our_Activities/Space_Engineering_Technology/Slam_dunk_for_Andreas_in_space_controlling_rover_on_ground
- [29] „One Big Step for Man as Astronaut Controls Robot from ISS“. Võrgumaterjal. Vaadatud 18. mai 2019. <https://phys.org/news/2015-09-big-astronaut-robot-space.html>
- [30] Fritsche, L., Unverzag, F., Peters, J. ja Calandra, R. 2015. „First-person tele-operation of a humanoid robot“. 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), 997–1002. Võrgumaterjal. Vaadatud 17. mai 2019. <https://doi.org/10.1109/HUMANOIDS.2015.7363475>
- [31] „iCub.org - an open source cognitive humanoid robotic platform“. Võrgumaterjal. Vaadatud 17. mai 2019. <http://www.icub.org/>
- [32] Rueckert, Elmar, Lioutikov, Rudolf, Calandra, Roberto, Schmidt, Marius, Beckerle, Philipp ja Peters, Jan. „Low-Cost Sensor Glove with Force Feedback for Learning from Demonstrations Using Probabilistic Trajectory Representations“. 2015. Võrgumaterjal. Vaadatud 17. mai 2019. <http://arxiv.org/abs/1510.03253>
- [33] „Oculus Homepage“. Võrgumaterjal. Vaadatud 17. mai 2019.
<https://www.oculus.com/>

- [34] „Kinect - Windows app development“. Võrgumaterjal. Vaadatud 27. aprill 2019.
<https://developer.microsoft.com/en-us/windows/kinect>
- [35] „NAO the humanoid robot | SoftBank Robotics EMEA“. Võrgumaterjal.
Vaadatud 17. mai 2019. <https://www.softbankrobotics.com/emea/en/nao>
- [36] „Recognition of Human Motions for Imitation and Control of a Humanoid Robot
- IEEE Conference Publication“. Võrgumaterjal. Vaadatud 17. mai 2019.
<https://ieeexplore.ieee.org/document/6363341>
- [37] „ROS/Introduction - ROS Wiki“. Võrgumaterjal. Vaadatud 29. oktoober 2019.
<https://wiki.ros.org/ROS/Introduction>
- [38] „ROS.Org | About ROS“. Võrgumaterjal. Vaadatud 29. oktoober 2019.
<http://www.ros.org/about-ros/>
- [39] „Documentation - ROS Wiki“. Võrgumaterjal. Vaadatud 29. oktoober 2018.
<https://wiki.ros.org/>
- [40] „Nodes - ROS Wiki“. Võrgumaterjal. Vaadatud 25. aprill 2019.
<http://wiki.ros.org/Nodes>
- [41] „Messages - ROS Wiki“. Võrgumaterjal. Vaadatud 25. aprill 2019.
<http://wiki.ros.org/Messages>
- [42] „ROS/Concepts - ROS Wiki“. Võrgumaterjal. Vaadatud 29. oktoober 2019.
<https://wiki.ros.org/ROS/Concepts>
- [43] „Topics - ROS Wiki“. Võrgumaterjal. Vaadatud 25. aprill 2019.
<https://wiki.ros.org/Topics>
- [44] „Services - ROS Wiki“. Võrgumaterjal. Vaadatud 25. aprill 2019.
<https://wiki.ros.org/Services>
- [45] „STAIR“. Võrgumaterjal. Vaadatud 29. oktoober 2018.
<http://stair.stanford.edu/index.php>
- [46] Quigley, Morgan ja Berger, Eric. „STAIR: Hardware and Software Architecture“. AAI 2007. Võrgumaterjal. Vaadatud 15. mai 2019.
<http://www.aaai.org/Papers/Workshops/2007/WS-07-15/WS07-15-008.pdf>
- [47] „ROS.Org | History“. Võrgumaterjal. Vaadatud 29. oktoober 2019.
<https://www.ros.org/history/>

- [48] „Distributions - ROS Wiki“. Võrgumaterjal. Vaadatud 29. oktoober 2019.
<http://wiki.ros.org/Distributions>
- [49] „gazebo_ros_pkgs - ROS Wiki“. Võrgumaterjal. Vaadatud 30. aprill 2019.
http://wiki.ros.org/gazebo_ros_pkgs
- [50] „Gazebo“. Võrgumaterjal. Vaadatud 30. aprill 2019. <http://gazebosim.org/>
- [51] „Leap Motion“. Leap Motion. Võrgumaterjal. Vaadatud 03. detsember 2018.
<https://www.leapmotion.com/>
- [52] „API Overview — Leap Motion C++ SDK v2.3 documentation“. Võrgumaterjal.
Vaadatud 25. aprill 2019.
https://developer-archive.leapmotion.com/documentation/v2/cpp/devguide/Leap_Overview.html
- [53] „How Does the Leap Motion Controller Work?“ 2014. Leap Motion Blog. 9. august 2014. Võrgumaterjal. 25. aprill 2019.
<http://blog.leapmotion.com/hardware-to-software-how-does-the-leap-motion-controller-work/>
- [54] „Leap Motion Teardown - learn.sparkfun.com“. Võrgumaterjal. Vaadatud 27. aprill 2019. <https://learn.sparkfun.com/tutorials/leap-motion-teardown/>
- [55] Wang, Qifei, Kurillo, Gregorij, Ofli, Ferda ja Bajcsy, Ruzena. „Evaluation of Pose Tracking Accuracy in the First and Second Generations of Microsoft Kinect“. 2015 International Conference on Healthcare Informatics. Võrgumaterjal. Vaadatud 27. aprill 2019. <https://ieeexplore.ieee.org/document/7349715>
- [56] Guna, Jože Jakus, Grega, Pogačnik, Matevž, Tomažič, Sašo ja Sodnik, Jaka. 2014. „An Analysis of the Precision and Reliability of the Leap Motion Sensor and Its Suitability for Static and Dynamic Tracking“. Sensors 14 (2). lk 3702–3720. Võrgumaterjal. Vaadatud 27. aprill 2019. <https://doi.org/10.3390/s140203702>
- [57] „Index of /16.04“. Võrgumaterjal. Vaadatud 28. aprill 2019.
<http://releases.ubuntu.com/16.04/>
- [58] „kinetic - ROS Wiki“. Võrgumaterjal. Vaadatud 15. mai 2019.
<http://wiki.ros.org/kinetic>
- [59] „leap_motion - ROS Wiki“. Võrgumaterjal. Vaadatud 28. aprill 2019.
http://wiki.ros.org/leap_motion

- [60] „V2 Tracking“. Võrgumaterjal. Leap Motion Developer. Vaadatud 28. aprill 2019. <http://developer.leapmotion.com/sdk/v2>
- [61] „Tip: Ubuntu, Systemd and Leapd“. Leap Motion Community. Võrgumaterjal. Vaadatud 15. mai 2019. <https://community.leapmotion.com/t/tip-ubuntu-systemd-and-leapd/2118/3>
- [62] „universal_robot - ROS Wiki“. Võrgumaterjal. Vaadatud 28. aprill 2019. http://wiki.ros.org/universal_robot
- [63] „MoveIt Motion Planning Framework“. Võrgumaterjal. Vaadatud 28. aprill 2019. <https://moveit.ros.org/>
- [64] „jog_arm - ROS Wiki“. Võrgumaterjal. Vaadatud 28. aprill 2019. http://wiki.ros.org/jog_arm
- [65] ut-ims-robotics/keyboard_publisher. Robotics at IMS Lab. 2018. Võrgumaterjal. Vaadatud 28. aprill 2019. https://github.com/ut-ims-robotics/keyboard_publisher
- [66] ut-ims-robotics/ur5_jog_arm. Robotics at IMS Lab. 2019. Võrgumaterjal. Vaadatud 28. aprill 2019. https://github.com/ut-ims-robotics/ur5_jog_arm
- [67] ut-ims-robotics/jog_arm. Robotics at IMS Lab. 2018. Võrgumaterjal. Vaadatud 28. aprill 2019. https://github.com/ut-ims-robotics/jog_arm
- [68] willcbaker/ur_modern_driver. „The new driver for the UR3/UR5/UR10 robot arms from universal robots“. Baker, Will. 2018. Võrgumaterjal. Vaadatud 15. mai 2019. https://github.com/willcbaker/ur_modern_driver
- [69] „UR5 collaborative robot arm | flexible and lightweight robot arm“. Võrgumaterjal. Vaadatud 28. aprill 2019. <https://www.universal-robots.com/products/ur5-robot/>
- [70] „[Gazebo_ros_control] Strange Behaviour When Adding Gazebo_ros_control to a Robot · Issue #612 · Ros-Simulation/Gazebo_ros_pkgs“. GitHub. Võrgumaterjal. Vaadatud 20. mai 2019. https://github.com/ros-simulation/gazebo_ros_pkgs/issues/612
- [71] „pepper_moveit_config - ROS Wiki“. Võrgumaterjal. Vaadatud 20. mai 2019. http://wiki.ros.org/pepper_moveit_config
- [72] „Humanizing Technologies“. Võrgumaterjal. Humanizing Technologies. Vaadatud 20. mai 2019. <http://humanizing.com>

Lisad

Lisa 1

Töö haldamiseks on loodud lahendus üles laetud GitHubi platvormi, kus leiab juhised selle kasutamiseks.

https://github.com/martinhallist/leap_teleoperations

Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Martin Hallist,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose „**Robotipõhine kaugkohalolu käte liigutuste ülekandmiseks**“, mille juhendaja on Karl Kruusamäe, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.
2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Martin Hallist

03.06.2019